

the cat canine search, the user clicks on one of the hyperlinks presented on the cat canine results page. In this embodiment, only the results page for the cat canine search is stored in the history, since the user's input (clicking on one of the results of that page) indicates that that particular results page was likely the most interesting and relevant to the user, since its results elicited a positive response from the user (e.g. clicking on one of the displayed results). Any intermediate results pages, such as that for the cat dog search, would be omitted from the history records of the browser. This may reduce cluttering the history list with unnecessarily duplicative, incomplete, or redundant entries, while preserving the results pages that are most likely to be relevant and of importance to the user. In one embodiment, a user may select whether to keep all intermediate searches in the history, or only those deemed likely to be relevant by the browser application, as described above.

[0069] In one embodiment, the functionality of the back button (e.g. 610 of FIG. 6A) of the browser may be configured similarly to that of the history function. A back button in a browser is typically used to return to the last page viewed. In one embodiment, after a search using the immediate search feedback aspect has been performed, the back button may be used to return to the last non-results page visited. For example, consider the situation where a user is browsing www.apple.com, then types ipod into the browser's integrated search field. In this example, three intermediate pages (those for i, ip, and ipo) may be presented to the user before the final complete ipod search results page is displayed. In one embodiment, clicking on the back button of the browser once will return the browser to the results page for the ipo search, since it was the last page visited. In an alternative embodiment, clicking on the back button of the browser once will return the browser to the www.apple.com web page, since it was the last page visited before the "ipod" search was initiated by the user.

[0070] FIG. 7 illustrates an embodiment of a user interface 700 for setting or adjusting various parameters relating to immediate search feedback, such as the temporal and substantive triggers. As described above, in one embodiment, certain aspects of the immediate search feedback may be adjusted or specified by a user through interaction with the user interface 700. The user interface 700 is displayed as a graphical user interface (GUI) object. Examples of GUI objects include windows (e.g. document, application, utility, dialog and alert windows), sheets, lists, menus (e.g. popup menus, pull-down menus), title bars, menu bars, toolbars, taskbars, palettes, clipboards, docks, and drawers (a child window that slides out from a parent window), among others. The user-adjustable parameters illustrated in the user interface 700 are not exhaustive; it will be appreciated that any of the various triggers, thresholds, and parameters discussed above may be set, specified, or adjusted by a user through a graphical user interface, such as for example that illustrated in FIG. 7. Furthermore, the specific user-manipulable control elements (e.g. buttons, sliders, check boxes, etc.) in FIG. 7 as they relate to specific parameters of the immediate search are merely illustrative of one embodiment of the invention; use of various other control elements are contemplated for controlling different parameters. Examples of such user-manipulable control elements include buttons, selection controls, sliders, radio buttons, checkboxes, menus, pop-up menus, pull-down menus, fields, text input fields, selection fields or other input fields, among others.

[0071] In one embodiment, the user interface 700 is presented as a window with control elements therein. In one embodiment, the user interface 700 includes a slider control 702 to adjust the time delay before a query is automatically submitted based on the received input (i.e. the temporal trigger). The slider control 702 allows a user to choose from a continuous range of allowable values. For example, in one embodiment, a user may manipulate a slider 704 to adjust the time delay between about 0 seconds and about 3 seconds. Other ranges of time may be used with other embodiments. In another embodiment, a user may specify a number of input characters, after which a query is automatically submitted. For example, a user may input a number into field 706, such as 4; in such a case, after four characters are input into a search field (such as search field 504 of FIG. 5A), a search query is automatically submitted. In one embodiment, various checkboxes (or other control elements, such as radio buttons) may be used to allow a user to control other aspects of the immediate search feedback, such as for example whether to turn the auto-complete feature on 708 (as described above), whether to delay a search until the input matches a recognized complete word 710 (as described above), whether to append wildcards to automatically submitted search query strings 712 (as described above), and whether to automatically adjust the automatic submission thresholds based on a connection speed 714 (as discussed above).

[0072] In one embodiment, an application programming interface (API) may be provided in a computing system to allow an application program to set or adjust the settings relating to immediate search feedback, such as the temporal and substantive triggers for automatically submitting a query. An API is an interface through which an application program can access an operating system of a computing device. For example, in one embodiment, a first software component may monitor a user's typing behavior/pattern to determine a timing parameter. For example, the first software component may monitor or record timing characteristics when a plurality of keys are actuated (e.g. the average time/delay/pause between successive input of text characters within a word processing application). Based on the observed typing pattern, a timing parameter is set. In one embodiment, for example, the timing parameter may be the average time elapsed between successive input of text characters. Through an API, the first software component may call a second software component (e.g. the operating system) to set a temporal threshold specified by the second software component. The temporal threshold may then be used to determine when to automatically submit a query to a search engine (i.e. the threshold specifies the duration of the pause when receiving text input, after which a search query may be automatically submitted).

[0073] The methods described above constitute computer programs made up of computer-executable instructions illustrated as blocks (acts) within the flow charts of FIGS. 3 and 4. Describing the methods by reference to a flow chart enables one skilled in the art to develop such programs including such instructions to carry out the methods on suitably configured computers (the processor of the computer executing the instructions from computer-readable media, including memory). The computer-executable instructions may be written in a computer programming language or may be embodied in firmware logic. If written in a programming language conforming to a recognized